

Instructors: Tunca Doğan, Fuat Akal, Aydın Kaya

TA: Hayriye Çelikbilek

Course: BBM103 Fall 2022

Subject: Loops, conditional statements, basic arithmetic operations, and functions.

Given: 03.11.2022

Due: 05.11.2022 23:00, to classroom.github.com

Quiz 2

The Aim of This Quiz

In this quiz, we expect you all to get practice on basic python commands, and also get experience with the python functions, programming environments, the user interface of the Integrated Development Environment (or IDE), python programming console, or terminal.

You should handle the problem in a single .py file and name it as “Quiz2.py”.

Problem1: Basketball Score Calculator

A basketball player in a game can score in 3 different ways. The player can

- (i) make a basket inside of the 3-point zone to score 2 points,
- (ii) make a basket outside of the 3-point zone to score 3 points,
- (iii) make a free throw to score a single point.

In this quiz, you are given how many 2-point, 3-point and free throws are made by a player. Your task is to calculate the player's overall score in that game using **command-line arguments**.

Specifications

- The input will consist of 3 variables, containing 2-point (int), 3-point (int) and free throws (int) made by a player.
- All input values will be integers.
- The output, an integer, will be the player's score.
- **Do not use input() function.**

- You must not import any modules.
- You cannot use conditional statements, loops, etc.
- You cannot define functions.
- You may use built-in functions.
- You should write, run and test your code in your local environment before uploading.
- In order to test your answer, you have to open a terminal inside the folder with your Quiz2 file, then you can call your Quiz2 script with different values for testing.
- The order of command-line arguments:

python3 Quiz2.py (i: 2 points) (ii: 3 points) (iii: single throws)

Sample Runs

```
> python3 Quiz2.py 10 3 5
```

```
34
```

```
> python3 Quiz2.py 5 4 8
```

```
30
```

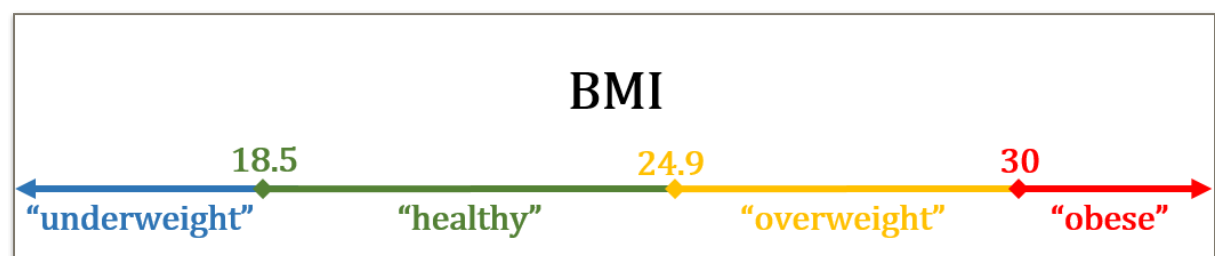
Problem2: Body Mass Index Calculator

In this question, you have to **define a function (do not call the function)** called `healthStatus(height, mass)` to determine a person's health condition by calculating his/her BMI (Body Mass Index). BMI can be calculated as follows:

$$\text{BMI} = \text{mass} / (\text{height} ** 2)$$

Specifications

- Function will take two float parameters called mass (kg) and height (m).
- You will first compute the BMI using mass and height.
- The returned value will be a string according to the BMI of the person:



- Your function should **return the result, without printing it.**
- **Do not use input() function.**

- Do not leave any code outside the function.
- Make sure that you comply with the function signature (name and parameters).
- You must not import any modules.
- You must not use iteration.
- You may use built-in functions.
- Remember that in this quiz, you have to define a function called `healthStatus(height, mass)`
- You should write, run and test your code in your local environment before uploading.
- In order to test your answer, you have to open a python3 interpreter and import your module named `Quiz2` then you can call your `healthStatus` function with different values for testing.

Sample Runs

```
>>> from Quiz2 import healthStatus
>>> healthStatus(1.83, 75)
'healthy'
>>> healthStatus(1.68, 80)
'overweight'
```

1. Grading

Evaluation	Points	Evaluate Yourself / Grading
• Using command-line properly	20	...
• Calculating basketball score properly	20	...
• Printing Results with string	-10	...
• Using try/catch (Bonus)	10	...
Problem1: Basketball Score Calculator Total	40	...
• Defining the <code>healthStatus</code> function properly	20	...
• Returning the result properly	10	...
• Calculating BMI properly	20	...
• Printing Results	-5	...
• Calling the function	-5	...
Problem2: Body Mass Index Calculator Total	50	...
Using conditional statements, loops etc. properly	8	
Writing down your name inside the script	2	...

Evaluation	Points	Evaluate Yourself / Grading
Any other mistake	-10	...
Quiz Total	100	...

2. Notes

- Do not miss the submission deadline.
- Save all your work until the quiz is graded.
- The quiz must be original, individual work. Duplicate or very similar quizzes are both going to be considered as cheating. Show any references or quotes for academic honesty.
- Write READABLE codes and their COMMENTS.
- You can ask your questions via [Piazza](#) and you are supposed to be aware of everything discussed on Piazza.
- You will use online [Github Classroom](#) system to submit your quiz.
- No other submission method (email or etc.) will be accepted. Do not submit any file via e-mail related with this quiz.
- You need to name your quiz as **a single file of “Quiz2.py”**.
- Do not forget to fill your grading table with your expected marks (this part is optional), and attach the table to your submission as “Quiz2.pdf”.
- Do not hesitate to ask questions to your TA. TA’s office hour will be by appointment.
- Do not forget to **write down your name and number as a comment inside** your code file (and inside your optional pdf file).
- You should write, run and **test** your code in your local environment before uploading.

3. Policy

All work on quizzes must be done individually unless stated otherwise. You are encouraged to discuss with your classmates about the given quizzes, but these discussions should be carried out abstractly. Discussions about a particular solution to a problem (either in actual code or pseudocode) will not be tolerated. In short, turning in someone else’s work (from the internet), in whole or part, as your own will be considered a violation of academic integrity. Please note that the former condition also holds for the material found on the web, as everything on the web has been written by someone else.